

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listings of Claims:

- 5402
1. **(Currently Amended)** A computer system for automatic synchronization of scripting variables between a page including action tags and a tag library, the computer system comprising:
- a page suitable for building an application with dynamic web content, the page including one or more action tags that are provided as text in a mark-up language;
 - a tag library;
 - a translator suitable for translating the action tags from the mark up language to an executable programming code that is executed at runtime to perform actions intended by the action tags;
 - a TagExtraInfo object for each action tag in the page, the TagExtraInfo object [including] providing a method that is accessed by the translator at translation time, the method [returns] returning, at translation time, information that includes a list of available scripting variables, and a variable type and scope associated with each scripting variable that is defined or modified by its associated action tag, thereby allowing the translator at translation time to use the information provided by the method to generate code that when executed at runtime will assign each of the scripting variables with appropriate runtime values with respect to the type and scope of each of the scripting variables;
 - a pageContext object for the page, the pageContext object including a runtime mapping of at least one scripting variable in the list of available scripting variables to a runtime value that is represented or can be represented in the tag library,
 - a tag handler that creates at runtime one or more objects that the page requires, the tag handler further operating to store the one or more created objects into the pageContext object; thereby allowing the one or more objects to be retrieved at runtime when the generated code is executed, the one or more objects being assigned at runtime to each of the scripting variables in the list of scripting variables that is returned by the method at translation time.

~~thereby allowing synchronization of the at least one scripting variable between the page and the tag library by using the value provided in the mapping.~~

2. **(Cancelled)**

3. **(Cancelled)**

4. (Previously Presented) The computer system of Claim 3, wherein the TagExtraInfo object comprises:

a valid object name for each variable;

a type for each variable; and

a scope parameter that specifies a variable's scope relative to the page.

5. **(Cancelled)**

6. (Previously Presented) The computer system of Claim 1, wherein the page is executed on a server that implements a container, and the page is converted to a platform independent code that is executed on the server.

7. **(Currently Amended)** A method for automatically synchronizing scripting variable between a page including one or more action tags and a tag library, the page suitable for building an application with dynamic web content, the one or more action tags being provided as text in a mark-up language which are translated at translation time to an executable code that is executed at runtime to synchronize the scripting variables at runtime, the method comprising:

instantiating, by a translator at translation time, for each action tag a TagExtraInfo object, the TagExtraInfo object providing a method that is accessed by the translator at translation time, the method capable of returning at translation time information that includes a list of available scripting variables and respectively associated variable types and scopes for each of the scripting variables in the list of available scripting variables, each of the scripting variables being defined or modified by an associated action tag;

invoking the method by a translator at translation time, wherein the invoking operates to pass a list of attributes associated with the one or more action tags;

receiving, as a result of the invoking of the method, a collection of returned TagExtraInfo objects, wherein each of the returned TagExtraInfo objects includes an available scripting variable, its variable type, and its scope in the page;

generating by the translator, based on the returned TagExtraInfo objects, executable code that is executed at runtime, wherein the executable code accesses at runtime data that will be stored in a pageContext object at runtime, the runtime data including appropriate runtime values for each of the available scripting variables;

storing at runtime, into the pageContext object, by a tag handler at runtime, one or more objects that the page requires, thereby allowing the objects for the one or more objects to be retrieved at runtime and be assigned at runtime to the list of scripting variables;

executing, at run time, the code generated by the translator to assign appropriate runtime values which are stored in the pageContext object to each of the scripting variables in the returned TagExtraInfo objects, thereby allowing the runtime values to be retrieved and assigned at runtime to each of the available scripting variables in the collection of returned TagExtraInfo objects.

~~A method for automatically synchronizing scripting variables between a page including one or more action tags and a tag library, the method comprising:~~

~~—— creating for each action tag included in the page a TagExtraInfo object that includes a list of available scripting variables and a variable type associated with each scripting variable that is defined or modified by its associated action tag;~~

~~—— translating the page by referring to the list of scripting variables in the TagExtraInfo object associated with each action tag in the page;~~

~~—— executing the page;~~

~~—— creating for the page at execution, a pageContext object that includes a mapping of scripting variables to values that are or can be represented in the tag library; and~~

~~—— synchronizing the scripting variables between the page and the tag library by using the values that are provided in the mapping of the PageContext object.~~

8. (Previously Presented) The method of Claim 7, wherein the TagExtraInfo object comprises:

a valid object name for each variable;

a type for each variable; and

a scope parameter that specifies a variable's scope relative to the page.

9. (Cancelled)

10. **(Currently Amended)** The method of Claim **[[1]] 7**, wherein the page is executed on a server that implements a container, and the page is converted to platform independent code that is executed on the server.

11. (Cancelled)

12. (Cancelled)

13. **(Currently Amended)** A computer readable media including computer program code for automatically synchronizing scripting variables between a page including one or more action tags and a tag library, the computer readable media comprising:

computer program code for a page suitable for building an application with dynamic web content, the page including one or more action tags that are provided as text in a mark-up language;

computer program code for a tag library;

computer program code for a translator suitable for translating the action tags from the mark up language to an executable programming code that is executed at runtime to perform actions intended by the action tags;

computer program code for a TagExtraInfo object for each action tag in the page, the TagExtraInfo object providing a method that is accessed by the translator at translation time, the method returning, at translation time, information including a list of available scripting variables, and a variable type and scope associated with each scripting variable that is defined or modified by its associated action tag; thereby allowing the translator, at translation time, to use the list to generate code that when executed at runtime will assign each of the scripting variables with appropriate runtime values with respect to the type and scope of each of the scripting variables

computer program code for a pageContext object for the page, the pageContext object including a runtime mapping of at least one scripting variable in the list of available scripting variables to a runtime value that is represented or can be represented in the tag library;

computer program code for a tag handler that creates at runtime one or more objects that the page requires, the tag handler further operating to store the one or more created objects into the pageContext object, thereby allowing the one or more objects to be retrieved at runtime when the code is executed, and the one or more objects being assigned at runtime to each of the scripting variables in the list of scripting variables that was returned by the method at translation time.

computer program code for creating for each action tag included in the page a TagExtraInfo object that contains a list of available scripting variables and a variable type associated with each variable that is defined or modified by its associated action tag;

—— computer program code for translating the page by referring to the list of scripting variables in the TagExtraInfo object associated with each action tag in the page;

—— computer program code for executing the page; and

—— computer program code for creating for the page at execution a pageContext object that contains a mapping of scripting variables to values that are or can be represented in the tag library; and

—— computer program code for synchronizing the scripting variables between the page and the tag library by using the values that are provided in the mapping of the PageContext object.

14. (Previously Presented) A computer readable medium as recited in claim 13, wherein the TagExtraInfo object comprises:

a valid object name for each variable;

a type for each variable; and

a scope parameter that specifies a variable's scope relative to the page.

15. (Previously Presented) A computer readable medium as recited in claim 13, wherein the page is converted to a first programming code which is different than a second programming code that is used to implement the tag library.

16. (Previously Presented) A computer readable medium as recited in claim 13, wherein the page is executed on a server that implements a container, and the page is converted to a platform independent code that is executed on the server.

17. (Previously Presented) A computer system as recited in claim 1, wherein the page is converted to a first programming code which is different than a second programming code that is used to implement the tag library.

18. (Previously Presented) A method as recited in claim 7, wherein the page is written in a first programming code which is different than a second programming code that is used to implement the tag library.